第27卷 第3期
2001年5月

自 动 化 学 报
ACTA AUTOMATICA SINICA

Vol. 27, No. 3
May, 2001

短文

支持虚拟企业建立的项目优化调度算法[1)]

毛 宁 陈庆新 陈 新

(广东工业大学机电系 广州 510090)

摘 要 研究敏捷制造模式下,动态企业联盟的形成过程中出现的多模式资源受限项目调度问题.与前人研究的问题有所不同,文中考虑了项目中每个任务对可更新(再生)资源需求的任意分布、可更新(再生)资源的最大供给量随时间而变化的情形.

关键词 多模式,资源受限,项目调度,不可中断,分枝定界.

A PROJECT SCHEDULING ALGORITHM FOR THE FORMATION OF VIRTUAL ENTERPRISES

MAO Ning CHEN Qing-Xin CHEN Xin

(*Department of Mechanical & Electronic Engineering, Guangdong University of Technology, Guangzhou* 510090)

Abstract This paper deals with the multi-mode multiple resource-constrained project scheduling problem (MRCPSP), which is abstracted from modeling the formation process of virtual enterprises of agile manufacturing in the next century. Different from the problem investigated by other researchers, the problem handled in this paper is of variable resource requirement and renewable resource availability constraints. As an extension to the famous DH-procedure, the branch-and-bound algorithm in this paper can solve this kind of most general project scheduling problems with optimality.

Key words Multi-mode, resource-constrained, project scheduling, nonpreemptive, branch-and-bound.

1 引言

多模式资源受限项目调度问题(multi-mode multiple resource-constrained project scheduling problem, MRCPSP)在动态企业联盟的形成过程中有着重要应用价值.动态企业联盟产生的原因之一是企业联合向市场投入新产品.在新产品的开发、生产和销售项目

1) 国家"八六三"/CIMS 跟踪(863-511-9843-008, 863-511-9944-008)和广东省自然科学基金(970380)资助项目.
收稿日期 1999-02-08 收修改稿日期 1999-10-08

规划中,针对每个任务,不同的潜在合作伙伴往往采用不同的资源(人和设备)和不同的工艺方案,这就导致了不同的任务工期;这些信息在合作伙伴项目任务投标书中已经明确.此外,由于每个潜在的合作伙伴同时还承担着其他任务,所以其可更新资源的最大供给量往往随着时间而发生变化.甚至几个合作伙伴还可能共享一些有限的关键资源等等.因此,针对同样的项目,不同的企业联盟必然会导致不同的项目执行时间和成本.那么项目核心企业有必要在确定企业联盟方案之前,对各种可能的潜在联盟方案进行项目虚拟调度,并利用结果对企业联盟的最后确定提供决策支持.

2 数学模型

首先需要解释一些常用符号的含义以方便模型描述.项目中的任务总个数为 J,可更新(再生)资源集为 R,不可更新(再生)资源集为 N,资源 r 在基本时间段 $(t-1,t]$ 的有限能力为 $K_{rt}\geqslant 0$,任务 j 可供选配的模式个数为 M_j(如有 M_j 个企业投标承担任务 j),模式 m 唯一地决定了任务 j 的工期 d_{jm}(如每个潜在的承担任务 j 的合作伙伴 m 承诺的任务工期为 d_{jm}),对应的每个任务对不可更新(再生)资源的消耗量为 $k_{jnr}\geqslant 0$;如果 s_{jm} 为任务 j 在模式 m 下的开工期,则 $w_{jnr\tau}\geqslant 0$ 为其在基本时间段 $(\tau-1,\tau]$ 上对可更新(再生)资源 r 的需求分布 $\tau=s_{jm},\cdots,s_{jm}+d_{jm}-1$,而在其它时间段上 $w_{jnr\tau}\equiv 0$.任务 j 的紧前任务集为 P_j,按照拓扑排序规则,针对 $\forall\, i\in P_j$,满足条件 $i<j$;对于具有多种可选配模式的任务,我们以其工期的非降次序排列模式;针对项目总工期的一个上界 T,可以按照以下方式来计算每个任务 j 的最早紧前可行完工期 EFT_j 和最迟紧前可行完工期 LFT_j.首先给每个任务配置相应工期最短的模式,再利用传统的前向递归计算每个任务的 EFT_j;然后,设置 $LFT_J=T$ 并利用传统的反向递归计算每个任务的 LFT_j.引入二位式决策变量 x_{jmt},$1\leqslant j\leqslant J$,$1\leqslant m\leqslant M_j$,如果任务 j 在模式 m 下于基本时间段 $(t-1,t]$ 结束时完成,它就等于1,而在其它情况下它都等于0.此外,对于网络源点1,假设 $d_{1m}=0$,$k_{1mr}=0$,$M_1=1$,$P_1=\varnothing$,$1\leqslant m\leqslant M_1$,$r\in R\cup N$;而对于网络汇点 J,也假设 $d_{Jm}=0$,$k_{Jnr}=0$,$M_J=1$,$1\leqslant m\leqslant M_J$,$r\in R\cup N$.因此,在文献[1,2]中模型的基础上,建立了具有任务资源需求分布与资源时变约束的一般 MRCPSP 0-1整数规划模型如下:

$$\sum_{m=1}^{M_j}\sum_{t=EFT_j}^{LFT_j} x_{jmt}=1,\quad j=1,\cdots,J, \tag{1}$$

$$\sum_{m=1}^{M_i}\sum_{t=EFT_i}^{LFT_i} t x_{imt}\leqslant\sum_{m=1}^{M_j}\sum_{t=EFT_j}^{LFT_j}(t-d_{jm})x_{jmt},\quad j=2,\cdots,J,i\in P_j, \tag{2}$$

$$\sum_{j=2}^{J-1}\sum_{m=1}^{M_j} w_{jmrt}\sum_{\varepsilon=t}^{t+d_{jm}-1} x_{jm\varepsilon}\leqslant K_{rt},\quad r\in R,\quad t=1,\cdots,T, \tag{3}$$

$$\sum_{j=2}^{J-1}\sum_{m=1}^{M_j} k_{jmr}\sum_{t=EFT_j}^{LFT_j} x_{jmt}\leqslant K_r,\quad r\in N, \tag{4}$$

$$x_{jmt}\in\{0,1\},\quad j=1,\cdots,J,\quad m=1,\cdots,M_j. \tag{5}$$

针对约束集(1)~(5)的常用目标函数是最小化项目总工期

$$\min\phi(M,S) \triangleq \sum_{t=EFT_j}^{LFT_j} tx_{JM_j}t. \tag{6}$$

此外,在实际应用中,往往还会出现这样的情况,即针对某些任务 j,要求其必须在时间区间$[g_j,h_j]$内完成,其中 g_j 和 h_j 分别是任务 j 的准备就绪时间和必须完工时间.这时,还需要算法就这种针对任务的时间约束,对产生的调度方案进行附加的可行性验证.问题(1)~(6)是最为一般和最为困难的(项目)调度问题之一,作为众所周知的 Job Shop 问题的一种推广,它属于一类 NP 困难的问题类[3].

3　算法

第一步.初始化.

设置 $T=999\ 9$作为项目总工期的一个上界,分枝定界搜索树的层数 $p=0$,初始化 $t=0$.设置 $t'=0$(t'是一个特定的决策点,以便于检验左移支配规则——最优解必要条件——的可应用性).针对所有指定了准备就绪时间 g_j 或完工期 h_j 的任务 j,输入 g_j 或 h_j.调度初始任务 $f_{11}=0$,$PS=\{1\}$以及 $S=\{1\}$;针对每个任务,配置相应工期最短的模式,计算第0层的下界 $LB(0)=q_1$;将所有以任务1为唯一紧前的任务放入割集 $C=\{x|x$ 具有唯一的紧前任务1$\}$,同时设置割集任务 x 的最早开工期 $s_x=g_x$.计算并存储不可更新(再生)资源的约束范围,并转向第二步.

第二步.模式选择与配置.

如果割集内没有尚未配置执行模式的任务,则转向第三步.

更新搜索树的分枝层 $p=p+1$.同时设置层类型标志 $flag(p)=0$.在目前不可更新(再生)资源的约束范围内,扣除不属于 C 的未调度任务对不可更新(再生)资源的最小所需量,再针对割集 C 中尚未配置执行模式的所有任务,确定所有可行的模式配置,构成割集任务的可行模式配置集 $MA(p)$,其中每个元素对应着割集 C 中尚未配置执行模式的所有任务.在目前剩余的不可更新(再生)资源约束条件下,各自选配一种相应的执行模式,总体构成的一种模式选配方案:如果割集任务的可行模式配置集 $MA(p)=\varnothing$,则转向第七步(回溯).

针对割集任务组合的每个可行执行模式配置方案,确定任务组合相应的剩余关键路径,选择具有最小关键路径的任务组合可行执行模式配置方案作为下界 $LB(p)$,如果 $LB(p)\geqslant T$,转向第七步(回溯).

给任务组合配置具有最小关键路径的可行执行模式方案,并将选中的配置模式从可行配置集 $MA(p)$中除去;存储以前的不可更新(再生)资源的约束范围,再计入当采用这一目前的配置方案时对不可更新(再生)资源的消耗,更新目前的不可更新(再生)资源的约束范围,并转向第三步.

第三步.时间的增加.

如果最后结束的任务 J 属于部分调度方案 PS,就对每个任务 j 以及配置的相应模式 m,验证是否满足开工期与完工期约束,即是否 $s_{jm}\geqslant g_j$,$s_{jm}+d_{jm}\leqslant h_j$.若是,则调度过程圆满完成;否则转向第七步(回溯).

如果改进了解,更新调度方案总长 $T=f_J$.如果 $T=LB(0)$,则已经获得了最优解,停

止;否则转向第七步(回溯).

在目前割集任务所配置的模式条件下,取以下三个量中的最小值,即所有割集任务的最早开工时间、可更新(再生)资源约束发生变化的下一个时刻,以及目前正进行任务对可更新(再生)资源的总需求发生变化的下一个时刻,作为下一个决策点 t. 针对所有任务 $j \in S$,并且 $f_{jm} \leqslant t$,更新正进行的任务集 $S=S \backslash \{j\}$.

检验割集支配性. 如果目前的割集受到支配,则转向第七步(回溯);否则暂存目前的割集 C,其中任务的最早开工时间与相应的执行模式、决策点 t,以及正进行的任务和它们的完成时间与相应的执行模式. 构造合格任务集,$E=\{x \in C \mid s_x=t, s_x \geqslant g_x\}$,并转向第四步.

第四步. 任务开工期或完工期的调度.

如果此时 $E=\varnothing$,则转向第二步.

将所有的合格任务放入正进行的任务集,针对所有的 $j \in E$,设置 $f_{jm}=t+d_{jm}$,$PS=PS \cup \{j\}$,以及 $S=S \cup \{j\}$. 更新割集 $C=C \backslash E \cup \{x \mid x$ 是任务 $j \in E$ 的一个紧后任务,而 x 的所有紧前任务都属于 $PS\}$,同时设置这些新的割集任务的最早可能开工期为 $s_x=\max \{\max \{f_{ym} \mid (y,x) \in P_x\}, g_x\}$. 针对每种可更新(再生)资源类型 r,在基本时间段 $(\tau-1,\tau]$ 上检验是否 $\sum_{j \in S} w_{jmr\tau} \leqslant K_{r\tau}, \tau=t+1, \cdots, \xi, \forall\ r \in R$,其中 $\xi=\max_{j \in S} \{f_{jm}\}$. 如果至少存在一种可更新(再生)资源 r,在其中某个基本时间段 $(\tau-1,\tau]$ 上,正进行的所有任务对其的需求总和超过了资源最大供给量,我们就面临了一个资源冲突,转向第五步;否则转向第二步.

第五步. 可更新(再生)资源冲突(导致可更新(再生)资源冲突的类型、严重程度、及可加以选择的解决方案).

更新搜索树的分枝层 $p=p+1$,同时设置层类型标志 $flag(p)=1$.

针对每种可更新(再生)资源类型 r,确定为解决资源冲突而必须释放多少单位的资源,即针对每种资源类型 r,以及有关的时间区间 $(\tau-1,\tau]$,计算 $c_{r\tau}=\sum_{j \in S} w_{jmr\tau}-K_{r\tau}$,其中 $t+1 \leqslant \tau \leqslant \xi, \forall\ r \in R, \xi=\max_{j \in S} \{f_{jm}\}$. 定义延滞集 $D(p)=\{D_q \mid D_q$ 是 S 的一个子集,针对所有可更新(再生)资源类型 $r, \forall\ r \in R, \sum_{j \in D_q} w_{jmr\tau} \geqslant c_{r\tau}$,其中 $t+1 \leqslant \tau \leqslant \xi, \xi=\max_{j \in S} \{f_{jm}\}$,$D_q$ 又不包含另一个 $D_v \in D(p)$作为子集$\}$. 针对每个 $D_q \in D(p)$,从产生可更新(再生)资源冲突的第一个 τ 起,确定以下四个量中的最小值,即可更新(再生)资源约束发生下一个变化的时刻、目前集合 $S \backslash D_q$ 中任务的可更新(再生)资源总需求发生下一个变化的时刻,如果 D_q 延迟所有正进行任务的最早完成时间及紧前约束不属于 D_q 的所有未调度任务在所有可行模式下的最早可能完成时间,并将其赋予所谓的延迟点 w_q. 针对每个延迟选择 D_q,按照 w_q 与每个任务 $j \in D_q$ 的所有剩余关键路径 q_j 中最大值的和,来计算基于紧前约束的下界 L_q. 选择具有最小 L_b 的 $D_b \in D(p)$,下标 b 指明了在第 p 层中剩余的最好延迟选择. 更新延滞集 $D(p)=D(p) \backslash D_b$. 计算 $LB(p)=L_b$. 如果 $LB(p) \geqslant T$,减少分枝层数 $p=p-1$ 并转向第七步. 否则,存储任务完成时间 f_{jm} 以及相应的执行模式,部分调度方案 PS、正进行的任务集 S、割集 C 及其任务的最早可能开工期 s_x 以及相应的执行模式、决策点 t 和 t',并转向第六步.

第六步. 延迟(执行延迟选择).

在分枝上搜索一个新的节点.定义 DS 作为在时刻 t' 之前已经决定开工但目前决定必须延迟的任务集 $DS=\{j\in D_b \mid f_{jm}-d_{jm}<t'\}$.执行延迟选择 $PS=PS\backslash D_b$ 及 $S=S\backslash D_b$.更新割集 $C=C\cup D_b\backslash\{y \mid x\in D_b$ 并且 $(x,y)\in P_y\}$.针对所有的 $j\in D_b$,更新最早可能开工期 $s_{jm}=w_b$;针对每个任务 $j\in D_b$,在对应的模式下确定了其开工期 s_{jm},如果存在一个任务 $j\in D_b$,其对应的 $s_{jm}+d_{jm}\geqslant h_j$,则转向第七步(回溯).

如果 DS 非空,援引左移支配规则如下.如果延迟选择中的一个任务 j 在搜索树中以前的某一层上被选中,从而迫使其在时刻 t' 才成为合格任务,如果任务 j 在时刻 t' 就开工,同时如果延迟执行任务集合 DS 可在不导致资源冲突的情况下容许任务 j 左移,则这一调度方案就受到支配,转向第七步(回溯).

更新 $t'=w_b$,并转向第二步.

第七步.回溯.

如果分枝层数 $p=0$,则停止.

如果在这一层上没有遗留的选择(延迟选择或模式配置选择),即如果当 $flag(p)=1$ 时,$D(p)=\varnothing$ 或当 $flag(p)=0$ 时,$MA(p)=\varnothing$;设置 $p=p-1$,并重复第七步.

如果当 $flag(p)=1$ 时,在这一层上的遗留延迟选择中,选择具有最小下界 L_b 的延迟 $D_b\in D(p)$,并更新延滞集 $D(p)=D(p)\backslash D_b$.计算 $LB(p)=L_b$,如果 $LB(p)\geqslant T$,减少分枝层数 $p=p-1$,并重复第七步.

恢复任务完成时间 f_{jm} 以及相应的执行模式,部分调度方案 PS、正进行的任务集 S、割集 C 及其任务的最早可能开工期 s_x 和相应的执行模式、决策点 t 和 t'.转向第六步.

当 $flag(p)=0$ 时,以割集任务总工期最短或最长原则排列遗留的可供选配的执行模式,并将选中的配置模式从可行的执行模式配置集 $MA(p)$ 中除去.恢复任务完成时间 f_{jm} 以及相应的模式,部分调度方案 PS、任务集 S、割集 C 及其任务的最早可能开工期 s_x、决策点 t 和 t',不可更新(再生)资源的约束范围.转向第二步.

参　考　文　献

1　Kolisch R,Drexl A. Local search for nonpreemptive multi-mode resource-constrained project scheduling. *IIE Transactions*,1997,**29**:987～999

2　Demeulemeester E,Herroelen W S. A branch-and-bound procedure for the generalized multiple resource-constrained scheduling problem. *Operations Research*,1997,**45**:201～212

3　Blazewicz J,Lenstra J K,Rinnooy Kan A H G. Scheduling subject to resource constraints: classification and comlexity. *Discrete Applied Mathematics*,1983,**5**:11～24

毛　宁　1962年生,1988年获西安交通大学机械及流体控制专业硕士学位.现任广东工业大学机电学院副教授.在国内外核心刊物上发表论文20余篇,获国家教育部科技进步二等奖两项.

陈庆新　1963年生,1992年获西安交通大学系统工程专业博士学位.现任广东工业大学机电学院教授.在国内外核心刊物上发表论文30余篇,获国家教育部科技进步二等奖两项.

陈　新　1960年生,1995年获华中理工大学机械设计与制造专业博士学位.现任广东工业大学机电学院院长,教授,博士生导师.在国内外核心刊物上发表论文30余篇,获多项省部级科技进步二等奖.